Amendments to the Claims:

Please cancel Claims 1, 14 and 15 without prejudice.

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Canceled)

2. (Currently Amended) The system as recited in Claim 21 wherein the target microcontroller is installed on a pod.

3. (Currently Amended) The system as recited in Claim 21 wherein ~~the microcontroller is copied in an FPGA (field programmable gate array) of the ICE~~ said in circuit emulator base station includes a field programmable gate array (FPGA), and wherein said emulated target microcontroller is programmed into said FPGA.

4. (Currently Amended) The system as recited in Claim 21 wherein the first memory includes a first static random access memory (SRAM) and the second memory includes a second SRAM.

5. (Currently Amended) The system as recited in Claim 21 wherein the first memory ~~further~~ includes a first ~~plurality of~~ central processing unit (CPU)

register ~~values~~ and the second memory ~~further~~ includes a second ~~plurality of~~ CPU register ~~values~~.

6. (Currently Amended) The system as recited in Claim 21 ~~wherein the first CPU includes a first program counter and the second CPU includes a second program counter, and~~ wherein lock step of said microcontroller code execution ~~execution~~ is maintained by ~~maintaining the first program counter and the second program counter in lock step~~ synchronizing said target microcontroller and said emulated target microcontroller.

7. (Currently Amended) The system as recited in Claim 21 wherein a ~~content of said first memory and a content of said second memory are displayed for comparison for consistency when execution of the microcontroller code is halted~~ said target microcontroller is a production microcontroller.

8. (Currently Amended) The system as recited in Claim 21 wherein ~~a state of the first CPU and a state of the second CPU are displayed for comparison for consistency when execution of the microcontroller code is halted~~ said in circuit emulator base station includes a trace buffer.

9. (Currently Amended) A method for debugging microcontroller code comprising:

a) programming said microcontroller code into a target microcontroller including a first memory and into an in circuit emulator base station including a second memory, a trace buffer, and an emulated target microcontroller which is not identical to and emulates operation of said target microcontroller and

initializing ~~a~~ said first memory ~~of an ICE (in-circuit emulator)~~ and ~~a~~ said second memory ~~of a microcontroller with microcontroller test code~~;

b) executing the microcontroller ~~test~~ code on the target microcontroller and on the emulated target microcontroller ~~ICE~~ in lock step by executing ~~the~~ same instructions using ~~the~~ same clocking signals;

c) verifying lock step of said executing step ~~execution~~ by comparing content of the first memory and content of the second memory solely on occurrence of any one of said executing step encounters a breakpoint and said executing step halts;

d) if lock step of said executing step ~~execution~~ is not verified, reporting an error, saving an execution history using ~~a~~ said trace buffer ~~coupled to the ICE~~ and debugging said microcontroller code; and

e) if lock step of said executing step ~~execution~~ is verified, continuing execution of the microcontroller ~~test~~ code.

10. (Currently Amended) The method of Claim 9 further comprising:

locating an error within the microcontroller ~~test~~ code by tracing the execution history using the trace buffer.

11. (Currently Amended) The method of Claim 9 wherein said first memory includes a first central processing unit (CPU) register and wherein said second memory includes a second CPU register, wherein said method further comprisi~~n~~ges:

verifying lock step of said executing step ~~execution~~ by comparing ~~register~~ contents of the first CPU register ~~memory~~ and ~~register~~ contents of the second CPU register ~~memory~~.

12. (Currently Amended) The method of Claim 9 wherein ~~the ICE is implemented using an FPGA (field programmable gate array)~~ said in circuit emulator base station comprises a field programmable gate array (FPGA), and wherein said emulated target microcontroller is programmed into said FPGA.

13. (Currently Amended) The method of Claim 9 wherein the target microcontroller is a production microcontroller.

14-15. (Canceled)

16. (Currently Amended) The system as recited in Claim ~~15~~ 22 wherein the target microcontroller is installed on a pod.

17. (Currently Amended) The system as recited in Claim ~~15~~ 22 wherein ~~the microcontroller is copied in an FPGA (field programmable gate array) of the ICE~~ said in circuit emulator base station includes a field programmable gate array (FPGA), and wherein said emulated target microcontroller is programmed into said FPGA.

18. (Currently Amended) The system as recited in Claim ~~15~~ 22 wherein the first memory ~~further~~ includes a first ~~plurality of~~ central processing unit (CPU) register ~~values~~ and the second memory ~~further~~ includes a second CPU ~~plurality of~~ register ~~values~~.

19. (Currently Amended) The system as recited in Claim ~~15~~ 22 wherein ~~a content of said first memory and a content of said second memory are displayed~~

~~for comparison for consistency when execution of the microcontroller code is halted~~ said in circuit emulator base station includes a trace buffer.

20. (Currently Amended) The system as recited in Claim ~~15~~ 22 wherein the target microcontroller is a production microcontroller.

21. (New) A system for debugging microcontroller code, comprising:

a target microcontroller including a first memory;

an in circuit emulator base station including a second memory and an emulated target microcontroller which is not identical to and emulates operation of said target microcontroller, wherein said target microcontroller and said emulated target microcontroller execute said microcontroller code, wherein said microcontroller code execution occurs in lock step by executing same instructions using same clocking signals; and

an interface coupled to said target microcontroller and said in circuit emulator base station, wherein lock step of said microcontroller code execution is verified by comparing said first memory with said second memory solely on occurrence of any one of said microcontroller code execution encounters a breakpoint and said microcontroller code execution halts, and wherein a mismatch between said first and second memories initiates debugging said microcontroller code.

22. (New) A system for maintaining lock step execution of microcontroller code during debugging, comprising:

a target microcontroller including a first memory;

an in circuit emulator base station including a second memory and an emulated target microcontroller which is not identical to and emulates operation

of said target microcontroller, wherein said target microcontroller and said emulated target microcontroller execute said microcontroller code, wherein said microcontroller code execution occurs in lock step by executing same instructions using same clocking signals;

a computer system coupled to said in circuit emulator base station and controlling debugging of said microcontroller code; and

an interface coupled to said target microcontroller and said in circuit emulator base station, wherein lock step of said microcontroller code execution is verified by said computer system by comparing said first memory with said second memory solely on occurrence of any one of said microcontroller code execution encounters a breakpoint and said microcontroller code execution halts, and wherein a mismatch between said first and second memories initiates debugging said microcontroller code.